Instrument Deployment for Mars Rovers

L Pedersen[†1], M. Bualat, C. Kunz[1], S. Lee[1], R. Sargent[1], R. Washington[2], A. Wright[1]

NASA Ames Research Center

Moffett Field, CA 94035-1000, USA

Abstract

Future Mars rovers, such as the planned 2009 MSL rover, require sufficient autonomy to robustly approach rock targets and place an instrument in contact with them. It took the 1997 Sojourner Mars rover between 3 and 5 communications cycles to accomplish this. This paper describes the technologies being developed and integrated onto the NASA Ames K9 prototype Mars rover to both accomplish this in one cycle, and to extend the complexity and duration of operations that a Mars rover can accomplish without intervention from mission control

Introduction

Approaching science targets, such as rocks, and placing instruments against them to take measurements is the *raison d'être* of a planetary surface exploration rover, such as the planned 2009 Mars Science Laboratory (MSL) rover (Figure 1). This is necessary to acquire samples, determine mineralogy, obtain microscopic images and other operations needed to understand the planet's geology and search for evidence of past or present life. Significant science simply cannot be done with remote measurements only.
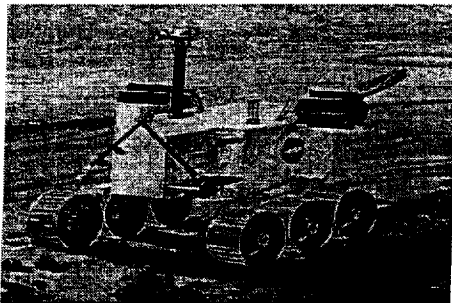

Figure 1 *Artist's conception of 2009 Mars Science Laboratory (MSL) rover. Current plans call for a nuclear powered vehicle operating for up to 1000 days. [JPL]*

Currently, a typical rover mission scenario starts when Mission Control uplinks a command sequence to the rover, specifying a detailed sequence of commands to take the rover to a particular target and deploy the desired instrument on it. The rover attempts to execute it as best it can, stopping when either the goal has been achieved or, as is more likely, conditions are such that the original sequence is no longer applicable. This could be due to obstacles in the way, navigation errors leading to loss of the target, excessive power use or unforeseen complexity of the target that prevents the instrument from being placed anywhere against it. Rover status and sensor data are downlinked to mission control at the next communications opportunity. Mission Control then assesses the situation and decides on the next command sequence to uplink. Several such command, or communications, cycles may be needed to accomplish the objective.

The light speed time delay between Earth and Mars varies between 10 and 20 minutes depending on their relative locations. Depending on the communications assets in place, only one such command cycle may be possible per Martian day, or sol.

This operating paradigm works well for spacecraft. Although far from benign, the space environment is very predictable and command sequences for a week's worth of activities are feasible. It does not work well for rovers in the complex environment of a planetary surface, even a relatively static one such as Mars.
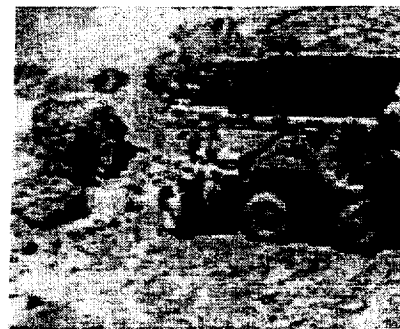

Figure 2 *Sojourner rover, observed from the Pathfinder lander on Mars in 1997. [JPL]*

The current *flight* state-of-the-art, the 1997 Sojourner Mars rover (Figure 2), requires at least 3 command cycles, each lasting a single sol, to accomplish the task of placing a relatively forgiving instrument on a compliant mounting against a rock several meters away. In addition, Sojourner could be observed by the Pathfinder lander, giving Mission Control a better view of the situation.

Reliability and verifiability are the fundamental concerns

† Email: pedersen@email.arc.nasa.gov

[1] Contractor to NASA with QSS Group, Inc.

[2] Contractor to NASA with RIACS

for flight missions and the reasons why Sojourner had such limited autonomy. The rover could only execute rigid command sequences, the default response to unexpected behavior was to abort the sequence and wait for the next communications opportunity. The reasons for this is that these rigid sequences could be rigorously checked and verified by mission control prior to being uploaded to the vehicle, guaranteeing that a whole class of failure modes would not occur.

Long delays of multiple sols to investigate each science target are unacceptable for a comprehensive study of a planetary surface. The technology to accomplish this objective in a single command cycle is essential. The 2009 MSL rover, as currently envisioned, *cannot* accomplish its science objectives without such a capability[1].

The MSL rover will operate far away from the landing craft. It will carry more sophisticated instruments than Pathfinder, and these must be placed against rock targets, up to 10m distant, with significantly greater precision.

At NASA's Ames Research Center (ARC), we are developing the robust autonomous instrument deployment capability needed for Mars rover missions. Our rover, K9, has demonstrated fully autonomous deployment of a microscopic camera against a rock in a relatively complex outdoor test environment (Figure 3).

Figure 3 *K9 rover approaches a rock target in the NASA Ames Marscape prior to autonomously placing the CHAMP microscopic camera against it using its 5 DOF robotic manipulator arm (August 2002).*

This paper describes our overall architecture and suite of technologies we are integrating to accomplish this; the K9 rover hardware and software pertinent to instrument placement and the results of our first demonstration of autonomous instrument placement.

System Architecture and Technologies

A complex sequence of activities is required for a rover to approach a target and place an instrument in contact with it (Figure 4). Currently, we primarily address the problem of instrument placement once the rover is at the target. However, for completeness, we begin with a review of methods for approaching the target.
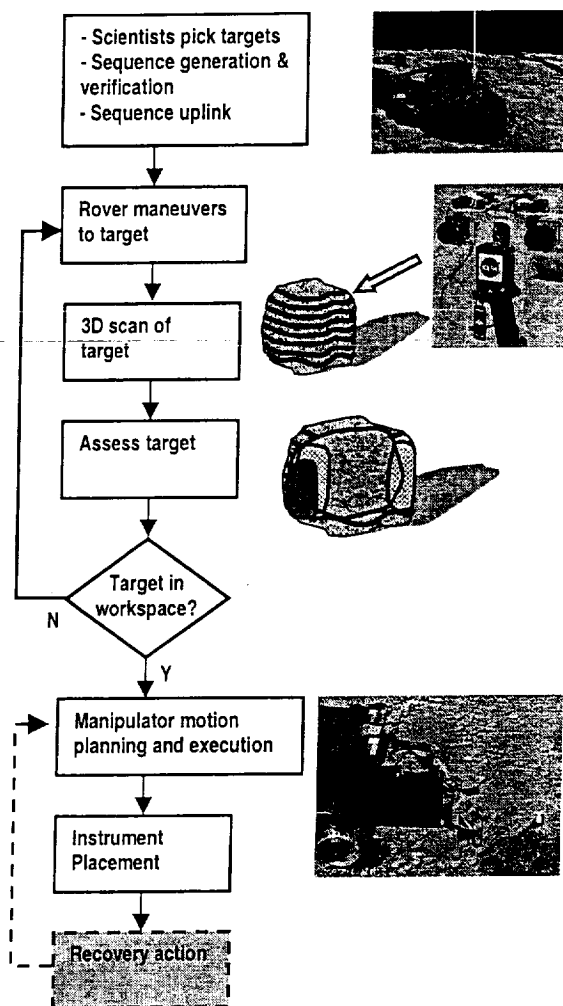

Figure 4 *Simplified sequence of operation that must be performed by a rover, such as K9, to autonomously place instruments on a target.*

Target Approach

First the rover must maneuver to within contact distance of the target. Because of navigation errors and uncertainty about the target location, the rover must keep track of the target location relative to itself throughout the maneuver. At the same time, it must avoid obstacles and pass through waypoints (if any).

Vision-based target tracking techniques can accomplish

this. Visual tracking is a closed-loop system that measures error directly through sensory feedback. As the rover moves, images are acquired of the presumed target area. Target features, such as 2D texture or 3D shape, are compared to features derived from the images and used to update the relative position of the target with respect to the rover. Provided target "lock" is maintained during the traverse, the target position relative to the rover can be obtained with relatively high accuracy. Initial uncertainties in target location, or those introduced by motion over unknown terrain, could potentially be eliminated by the visual tracking.

A 2D feature based visual servoing approach used on a previous rover at Ames, Marsokhod [2], relies on binary correlation to match features in the spatio-temporal image stream with features from a template image of the target. This is used to determine the target location in subsequent images acquired as the rover moves. Knowing the target location, a control loop keeps the rover navigation cameras foveated onto the target and directs the rover directly towards it.

Binary sign correlation is implemented using logical, rather than arithmetic operators; in a single processor exclusive OR (XOR) instruction on a 32-bit computer, 32 pixels can be compared in one instruction. The instruction level parallelism of this correlation approach makes it amenable to the limited processing requirements of near-term rover missions.

While fast, 2D appearance based techniques have a tendency to drift, and are not robust to the change in target appearance as the rover moves around or towards it. This is because they lack a sense of the 3D nature of the world.

The Rocky-7 3D stereo-based technique [3] uses shape information about the scene to supplement the information from a 2D feature tracker. Stereo-based shape tracking techniques are robust to noise and lighting variations but they are sensitive to calibration parameters and are computationally intensive.

Target Assessment and Instrument Placement

Once the rover has moved up to the target, it must determine where to place the instrument, what pose is needed, and check that the target surface will even permit the instrument to be placed there.

If Mission Control specified a particular final pose for the instrument, relative to a target that has been accurately tracked, then this task is unnecessary. The FIDO rover [4] demonstrated this. Using visual navigation techniques it can approach an exact target spot, localized from rover

the with 3cm precision. Once there, FIDO lowers an arm-mounted microscopic camera from a point directly above the target until a focused image is acquired. However, taking measurements from above targets is not always sufficient. Arbitrary instrument poses may be needed.

Scientists at Mission Control might wish to specify an entire rock as a target, not just a given point. Not only is such over-specification unnecessary; it may over-constrain the problem, and might not even be feasible prior to the rover approaching close enough to the rock to see it in sufficient detail. Alternatively, it might simply not be possible to track a single point with enough precision. In these cases, scientists are compelled to request a measurement anywhere on a rock (or large area on it).

The first step in determining where to place an instrument anywhere on a rock target (or other large area) is to obtain a 3D scan of the work area. This can be done with stereo cameras. It is important that they be well calibrated with respect to the rover manipulator arm, as the derived 3D point cloud will be used to compute desired instrument poses.

Next, the rock (or target area) in the 3D model of the work area must be segmented from the background. We have developed an iterative 3D clustering algorithm [5], based on the statistical EM algorithm, for this purpose. This algorithm is very robust to noise, requiring only that the ground be relatively flat (but at an arbitrary orientation) and the work area have at most one rock significantly larger than any clutter in the scene. If several large rocks are present in front of the rover, it becomes necessary for the workspace to be partitioned amongst them before applying this algorithm. Otherwise, it may aggregate several rocks together as a single rock or segment a random selection. Rocks piled up together will be aggregated.

The segmentation algorithm does not require many 3D points from the workspace to segment it. Therefore, the acquired 3D point cloud from the scene can be aggressively sub-sampled, enabling this algorithm to execute very rapidly.

Next, all points in the target area must be checked for consistency with the rover instrument to be placed. The simplest check for each point is to find all points within a given radius, compute the best-fit plane, and check the maximum deviations do not exceed some preset tolerance. The points are prioritized according to how flat the area is. Doing this also gets us the surface normal at each point in the target area. The result is a prioritized list of instrument positions and orientations (opposite to the

surface normals).

Finally, the instrument can be placed. First, via a series of pre-planned waypoints the arm is un-stowed and put in a holding position. Next it goes to a pose near the highest priority target pose in the workspace, holding back a safe distance along the target surface normal. To compensate for possible small errors in surface location, the instrument's final approach is along the measured normal to the target rock face, moving slowly forward until contact is confirmed by mechanical sensors.

Robust Execution and Resource Management

In order to accomplish the task of instrument placement within a single cycle with the robustness required for a mission, the on-board software must be able to handle failures and uncertainties encountered during the previously described component tasks. A task may fail, requiring recovery or retrying. Tasks may exhibit a high degree of variability in their resource usage, using more (or less) time and energy than expected. Finally, the state of the world and the rover itself may be predictable only to a limited extent. These factors require that the rover's software have the ability to reason about a wide range of possible situations and behaviors. A simple script is insufficient; instead, the rover can use either on-board task planning or off-board planning in conjunction with robust on-board execution.

We have chosen the approach of off-board planning along with robust on-board execution. This is more consistent with current mission practice, which requires intensive sequence verification before uplink. In addition, the perceived additional risk of an on-board planner could delay acceptance by mission managers.

Our approach is to use the Contingent Rover Language (CRL) along with the CRL Executive [6] for the on-board executive. The CRL Executive allows conditional branches to specify alternative plans of actions, libraries of "floating" contingency plans to handle situations that may occur at any time during plan execution, and utility-based decision-making to trade off alternatives with respect to science return.

K9 Rover

The K9 rover (Figure 3) is mechanically identical to the FIDO rover, itself an advanced technology rover that is a terrestrial prototype of the rover that NASA/JPL plans to send to Mars in 2003 (see http://fido.jpl.nasa.gov). K9's mobility sub-system consists of a six-wheel rocker-bogie suspension system and is capable of traversing over obstacles up to 30 cm in height.

The main CPU is a 750 MHz PC104+ Pentium III running the Linux operating system. An auxiliary microprocessor communicates with the main CPU over a serial port and controls power switching and other I/O processing. The motion/navigation system consists of motor controllers for the wheels, arm joints, and pan/tilt unit, a compass/inclinometer, and an inertial measurement unit.

The K9 rover software architecture uses the Coupled Layered Architecture for Robotic Autonomy (CLARAty) [7] developed at JPL, in collaboration with ARC and Carnegie Mellon University. By developing our instrument placement technology under the CLARAty architecture, we can easily port the system to other robots running CLARAty.

K9 Cameras

K9 is equipped with a front-mounted forward looking pair of b/w stereo hazard cameras and mast-mounted stereo pairs of high resolution color science cameras and wide field of view b/w navigation cameras (Figure 5). The navigation and science stereo camera pairs are mounted on a common pan-tilt unit, and can acquire image panoramas from around the rover.
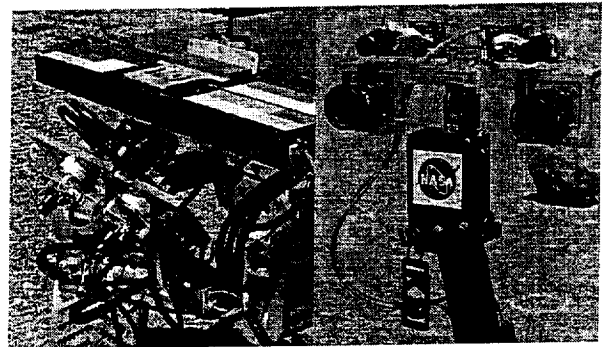

Figure 5 *K9 Stereo Hazard cameras (left) and Pan-Tilt mount with (right) with navigation cameras and high-resolution science camera stereo pairs.*

The hazard cameras overlook the arm workspace. Being fixed, and close to the target area, they are the easiest to calibrate with respect to the arm, and are therefore the current means for 3D scanning of the target area.

The hazard cameras are calibrated using a custom target mounted to the arm's end-effector (Figure 6) and designed such that each checkerboard intersection is uniquely identifiable by software. After taking several image pairs with different arm configurations and identifying the intersections in each image, we derive the camera intrinsic parameters, and an initial estimate of the extrinsic parameters using the OpenCV computer vision package. We then refine the extrinsic camera parameters,

as well as the estimate of the location of the target with respect to the end-effector, by adjusting the parameters while minimizing the total projection error over all the image pairs taken. The resulting model is a full characterization of the relationship between the two cameras, and between the cameras and the arm, so that stereo depth images taken with the cameras can be immediately used for arm positioning.
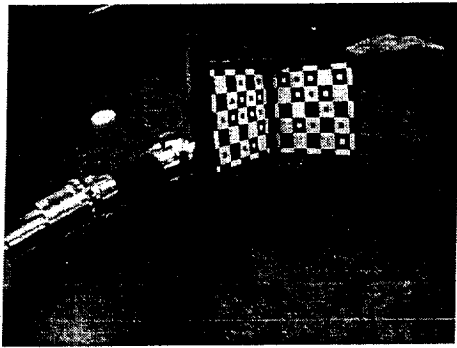

Figure 6 *Combined calibration of K9 front stereo hazard cameras and manipulator arm.*

Manipulator Arm

K9's instrument arm (Figure 7) is a 5-DOF robotic manipulator based on 4 DOF FIDO MicroArm IIA design from JPL [8]. It is approximately 5.0 kg with a total extended length of 0.79 meters. The waist yaw, shoulder pitch, elbow pitch, forearm twist (designed at Ames), and wrist pitch joints of the arm allow arbitrary x-y-z instrument placement as well as pitch and yaw control within the arm workspace. These rotational aluminum joints are connected by graphite epoxy tube links. The links are configured in a side-by-side orientation, with the two links running directly next to each other.
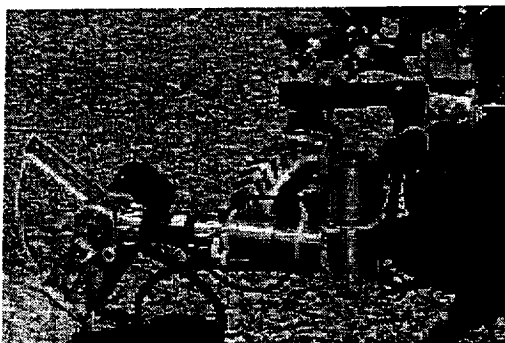

Figure 7 *K9 5 DOF manipulator arm, with CHAMP microscopic camera mounted at the end.*

The payload mass for K9's arm is estimated to be about 1.5 kg (3.3 lbs) with a strong-arm lifting capacity of about 2.5 kg (5.5 lbs) when fully extended in the horizontal position. Each joint in the arm has an embedded MicroMo 1319 series motor with an integrated planetary gear head and magnetic encoder. (Additional harmonic drive gearing was needed past the actuator to meet the significant torque requirements.) The no-load output speed varies from joint to joint, but averages about 0.1 radians per second. External to each joint is a multi-turn potentiometer that is coupled to the rotor and is used for initial arm calibration. The calibration procedure and magnetic encoders result in a positional accuracy of approximately 2 mm.

CHAMP Microscopic Camera

Affixed at the end of K9's arm is the CHAMP (Camera Hand-lens MicroscoPe) microscopic camera [9] (Figure 7). It has a movable CCD image plane, allowing it to obtain focused images over a wide depth of field, from a few millimeters up to several meters.

Because rotation about CHAMP's long axis does not need to be controlled, placing CHAMP flat against a rock requires control of five degrees of freedom. K9's arm has a full 5 degrees of freedom, removing the need to coordinate simultaneous arm and rover base motion. The rover's base only needs to move to within arm's reach of the rock and can remain stationary during arm movement.

CHAMP has three spring-loaded mechanical distance sensors around its face (Figure 12) that report contact with the rock. Because the rock surface is known to be flat, these three such sensors are sufficient for the final placement of the instrument After contact, these sensors can provide feedback necessary to fine-tune the instrument's distance and to correct any errors from the stereo surface normal measurement, although at the time of writing this final adjustment has not been implemented.

CHAMP can acquire a Z-stack of images from a target, each focused at a slightly different depth. These can be combined into a composite focused image or 3D mesh through a two step process: first, each pixel in each image is assigned a focus value corresponding to the sum of absolute differences among pixels within a small window around the pixel. The images are then registered to each other (this step is necessary because of wind and vibration, especially at extremely close range), using the phase shift correlation algorithm described in [10][11]. Finally, the pixels that are most in-focus down a given column in the stack are selected for the composite image (Figure 13). Using focus motor position information, each in-focus pixel can be projected into 3-space, allowing for the reconstruction of a 3D mesh.

Instrument Placement Demonstration

In August 2002, autonomous instrument placement was successfully demonstrated using a subset of the technologies described here (neither robust execution nor visual servoing were used). K9 approached a target from a distance of 2m, driving forward in a straight line using odometry and deduced reckoning (Figure 3).

The outdoor test site had moderate clutter, including scattered cobble and loose soil. The target rock itself is a complex aggregate of two rocks, one with a smooth surface and the other one grossly misshapen (Figure 8). Note the different textures and colors. The target and rover were oriented to minimize shadows in the hazard camera field of view.
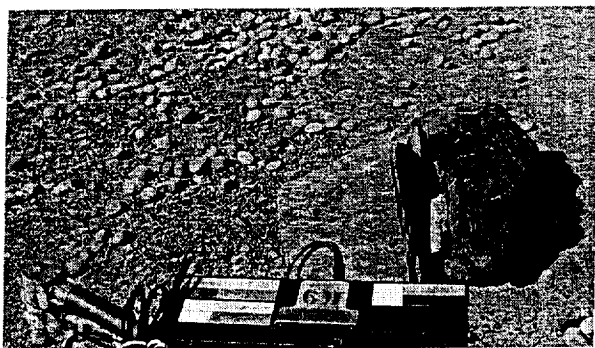

Figure 8 *August demo rock target scene. Rock targets are not in dead center of rover manipulator workspace due to rover navigation inaccuracies.*

Stereo images of the workspace were acquired with the hazard cameras, outfitted with neutral density filters to counteract the bright sunlight. These were processed to obtain the 3D model below.
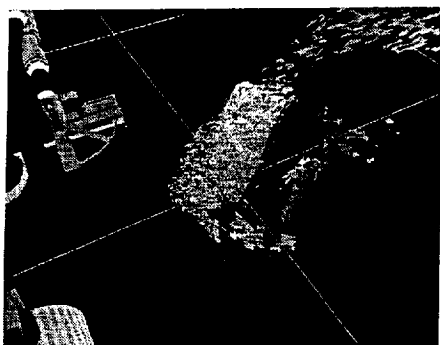

Figure 9 *K9 3D model of rock targets as rendered in Viz.*

The dot cloud from this 3D model was passed on to the rock/ground segmentation routine (Figure 10) and thence checked for areas consistent with CHAMP (Figure 11).

Finally, CHAMP was placed on the highest priority point,

which was on the flat surface, within the rover workspace (Figure 12). A Z-stack of microscopic images was then obtained (Figure 13), proving that the system can indeed autonomously place the CHAMP instrument and obtain measurements.
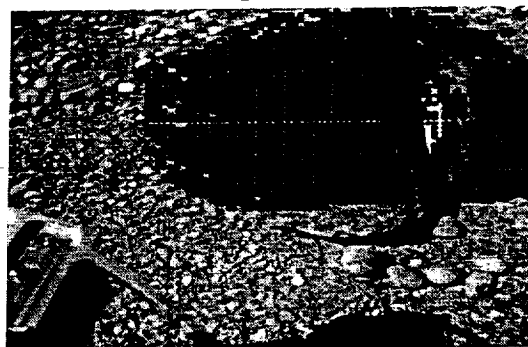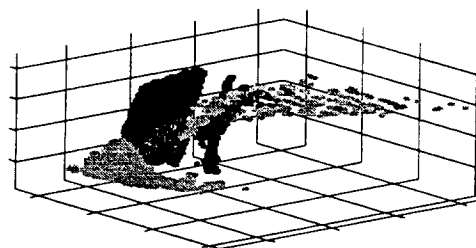

Figure 10 *Top: 3D rock and ground segmentation of 100x sub-sampled dot cloud from Figure 9. Bottom: rock points from above dot cloud superimposed on left stereo hazard camera image of work area. All rock points within 5 cm of ground plane are excluded to ensure instrument safety. Blank areas within rock are caused by missing data in the dot cloud (due to inadequate texture for stereo correlation in those areas).*

Figure 11 *Locations on aggregate rock surface determined to be consistent with the CHAMP microscopic camera. Points are prioritized according to flatness and amount of usable stereo data. For this demo, all rock points within a 5 cm radius had to be within 1 cm of the best-fit plane.*
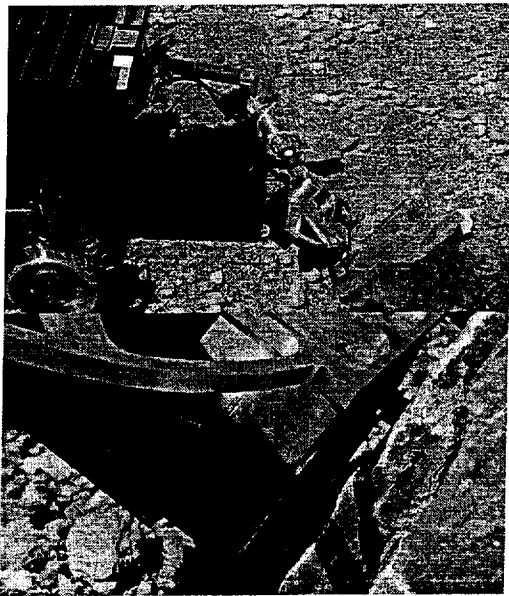
Figure 12 *Top: Final placement of CHAMP against the rock at the highest priority reachable point. Bottom: Close-up view of CHAMP showing the contact sensors pushed up against the target rock surface.*

Figure 13 *Focused composite of a Z-stack of CHAMP microscopic images obtained after final placement on the target rock. Image misalignments due to wind induced rover motion are automatically corrected. Maximum resolution is approximately 50 um per pixel, sufficient to show fine crystal structures.*

Future Work

Our next step is to incorporate the visual servoing technology under development at both Ames and JPL [2] [3][4]. This will enable K9 to autonomously keep track of a distant target as it approaches it, and brings it within the arm workspace.

Systematic end-to-end testing in realistic field environments is essential to make the system reliable enough. So far, modest reliability has been demonstrated in a relatively simple environment. Our final goal is a

very robust system capable of operating in a complex Martian environment that includes many rocks and significant clutter (Figure 14). Towards this, we are integrating our system with a simulation facility [12] and are planning field tests in both the Ames Marscape test facility and an undisclosed desert location.

Figure 14 *Mars rock scene, with significant clutter, few color variations and overlapping rocks of comparable sizes.*

We are upgrading the Ames Viz 3D virtual reality science interface [13], used for Pathfinder, for operators to select rock targets (Figure 15) and specify waypoints to get to them.

Viz gives users a virtual presence at the rover location, allowing scientists to explore a 3D virtual terrain generated by the Ames Stereo Pipeline software [14] using down linked stereo images of the site. Both Viz and Stereo Pipeline will be used by MER science teams in 2003.
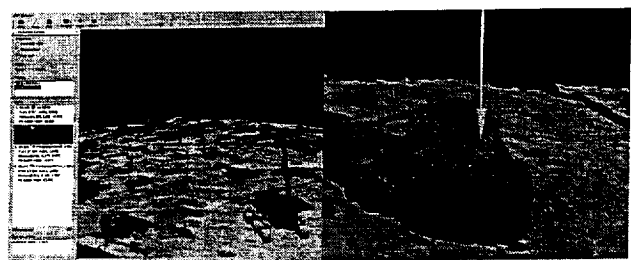

Figure 15 *The Viz immersive 3D virtual reality science interface used by scientists to study the 1997 Pathfinder landing site. We have upgraded it to allow users to specify rock targets for a rover.*

Once targets are selected, we can compute their positions and other information (such as template images) needed for a rover to go to them and place instruments. A CRL [6] rover execution sequence will be generated from this information, using a ground based limited incremental contingency planner under development [15]. Continued integration of the CRL Executive [6] with K9 will permit

the rover to execute this sequence. The sequence will permit flexible execution times and include conditional branches to recover from failures and visit multiple rocks as permitted by resources, such as power and time. This flexibility to deal with many possible contingencies will improve reliability and enable fully autonomous operations for longer durations before additional input from Mission Control is needed.

The integration of Viz, the contingent planner, conditional exec, K9 rover and instrument placement capabilities complete the set of components for end-to-end integrated demonstrations and realistic testing of our autonomous instrument placement capability and associated technologies for rover autonomy.

Conclusions

It has been speculated that the use of nuclear power to extend the 2009 Mars rover mission to 1000 days decreases the need for this kind of autonomy, as there would be sufficient time to accomplish measurements in the traditional, time consuming way, without having to risk autonomy. This is fallacious for several reasons. Over time the risk of a rover failure increases, hence it is important to get the baseline measurements as quickly as possible. The cost of operating a mission in the traditional manner, with a large co-located science and operations team for 1000 days is very high. In fact, it may not even be possible to obtain sufficient qualified personnel prepared to take time out from their careers to operate a rover for 3 years. Autonomy to alleviate this bottleneck is essential.

Ultimately, to fully explore an area to understand its geology and search for evidence of past or present life may require examining many hundreds, if not thousands, of rocks. Without automation, a few score rocks at most can be examined in a single mission.

This work demonstrates the eminent feasibility of autonomously, and robustly, placing science instruments against a rock target. Doing so dramatically increases the science return of future rover missions.

References

[1] Krasner, S.M., Tamppari, L., Steve Peters, S., Limonadi, D. (2002), "MSL Scenarios and Autonomy Requirements", MPSET meeting 4/11/2002.

[2] Wettergreen, D., H. Thomas, M. Bualat, (1997) "Initial Results from Vision-based Control of the Marsokhod Rover," in *proc. IEEE/RSJ International Converence on Intelligent Robots and Systems*, Grenoble. France, September 7-12, 1997.

[3] Nesnas, I., M. Maimone, H. Das (2000), "Rover

Maneuvering for Autonomous Vision-Based Dexterous Manipulation", *proc. IEEE International Conference on Robotics and Automation*, San Francisco, CA. 2000.

[4] Huntsberger, T., H. Aghazarian, Y. Cheng, E.T. Baumgartner, E. Tunstel, C. Leger, A. Trebi-Ollennu, and P.S. Schenker, (2002) "Rover Autonomy for Long Range Navigation and Science Data Acquisition on Planetary Surfaces", in *proc. IEEE International Conference on Robotics and Automation*, Washington, D.C. May 2002.

[5] Pedersen, L., (2002) "Science Target Assessment for Mars Rover Instrument Deployment," in *proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, September 30 – October 4, 2002.

[6] Bresina, J., K. Golden, D.E. Smith, R. Washington, (1999) "Increased Flexibility and Robustness for Mars Rovers", in *international Symposium on Artificial Intelligence, Robotics and Automation in Space*, 1999.

[7] Nesnas, I., R. Volpe, T. Estlin, H. Das, R. Petras, D. Mutz, (2001) "Toward Developing Reusable Software Components for Robotic Applications," in *IEEE/RSJ Internation Conference on Intelligent Robots and Systems*, 2001.

[8] Jet Propulsion Laboratory, "Planetary Dexterous Manipulators. MicroArm IIA and MastArm Specification Sheet."

[9] Lawrence, G.M., J.E. Boynton, *et al*, (2000), "CHAMP: Camera HAndlens MicroscoPe", in The 2nd MIDP Conference, Mars Instrument Development Program. JPL Technical Publication D-19508, 2000.

[10] Kuglin C., and D. Hines, (1975) "The Phase Correlation Image Alignment Method", in proc. *IEEE International Conference on Cybernetics and Society*, pp163-165, 1975

[11] Hill, L., (10 August 2001) http://www.ee.surrey.ac.uk/Personal/L.Hill/pc.html

[12] Flückiger, L. and C. Neukom, (2002), "A new simulation framework for autonomy in robotic missions," in proc. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, September 30 – October 4, 2002.

[13] Nguyen, L., M. Bualat, L. Edwards, L. Flueckiger, C. Neveu, K. Schwehr, M. Wagner, E. Zbinden (2001), "Virtual Reality Interfaces for Visualization and Control of Remote Vehicles," Autonomous Robots 11(1), 2001.

[14] Stoker, C., E. Zbinden, T. Blackmon, B. Kanefsky, J. Hagen, C. Neveu, D. Rasmussen, K. Schwehr, M. Sims (1999), "Analyzing Pathfinder Data Using Virtual Reality and Super-resolved Imaging," Journal of Geophysical Research, vol 104, no E4, pp. 8889-8906, April 25, 1999.

[15] R. Dearden, N. Meuleau, S. Ramakrishnan, D. Smith, and R. Washington, (2002), "Contingency Planning for Planetary Rovers," in *3rd Intl. NASA Workshop on Planning & Scheduling for Space*, 2002